CLAIMS

What is claimed is:

1. A method comprising:

assembling tokens that define processing for producing a graphical state;

assembling a shell rasterizer using the tokens;

selectively modifying portions of the shell rasterizer with replacement logic; and

selectively inserting memory management logic into portions of the shell rasterizer to produce a modified rasterizer.

2. The method of claim 1 further comprising associating a pointer reference to a location of the modified rasterizer in a hash table, wherein the pointer reference is hashed based on the graphical state.

3. The method of claim 1 further comprising returning a pointer reference after a subsequent request is received for the graphical state.

4. The method of claim 1 wherein selectively modifying the portions further comprises determining a machine architecture that processes the method in order to select the portions.

5. The method of claim 1 wherein selectively inserting the memory management logic includes inserting the memory management logic based on simulated memory states for an executing shell rasterizer.

6. The method of claim 5 wherein selectively inserting the memory management logic further includes inserting the memory management logic using a memory interface associated with memory management for a machine architecture that processes the method.

7. A method comprising:

selectively replacing rasterizer logic in a rasterizer based on an architecture of a machine that processes the rasterizer;

selectively inserting memory management logic into the rasterizer using properties of the architecture; and

indexing the modified rasterizer in memory.

8. The method of claim 7 wherein selectively replacing rasterizer logic includes varying the rasterizer logic based on the properties and a graphical state.

9. The method of claim 7 wherein selectively inserting the memory management logic includes selecting the memory management logic based on simulating the execution of the rasterizer on the architecture.

10. The method of claim 7 wherein selectively inserting the memory management logic includes adding pushing and popping instructions for adding and removing portions of the memory management logic from a stack.

11. The method of claim 7 wherein indexing the rasterizer includes hashing a pointer reference to the rasterizer, wherein the pointer reference is hashed based on a graphical state associated with the rasterizer.

12. The method of claim 7 further comprising returning the modified rasterizer or a pointer reference to the modified rasterizer after a request for the modified rasterizer is received.

13. An article having a machine accessible medium having associated instructions, wherein the instructions, when executed, produce a rasterizer, the machine comprising at least one component performing:

assembling a shell rasterizer from a provided graphical state;

modifying the shell rasterizer with replacement logic and memory management logic to produce a modified rasterizer; and

indexing the modified rasterizer based on the provided graphical state.

14. The article of claim 13 wherein the instructions further comprise flushing memory used for producing the modified rasterizer after indexing the rasterizer.

15. The article of claim 13 wherein the instructions further comprise selecting the replacement logic and the memory management based on properties of an architecture for the machine and the provided graphical state.

16. The article of claim 13 wherein the instructions further comprise acquiring the replacement logic from a library of routines associated with performing rasterizer operations.

17. The article of claim 13 wherein the instructions further comprise acquiring the memory management logic based on an application programming interface (API) library associated with a memory stack of a machine.

18. A system, comprising:

a token building application that assembles a processing order needed to produce a graphical state; and

a composing application that assembles a generic shell rasterizer to satisfy the processing order;

wherein the composing application also dynamically replaces and inserts logic into the shell rasterizer to produce a modified rasterizer.

19. The system of claim 18 further comprising an indexing application for associating the graphical state with a pointer reference to a location of the modified rasterizer.

20. The system of claim 18 wherein the composing application uses replacement logic selected for a specific machine architecture.

21. The rasterizer building system of claim 18 wherein the composing application uses insertion logic selected based on a stack interface associated with a machine architecture.

22. A data structure residing in a computer-accessible medium for producing a rasterizer image, the data structure comprising:

shell logic produced from a graphical state;

replacement logic that selectively replaces portions of the shell logic based on an architecture of a machine that will process the data structure; and

insertion logic that is selectively intertwined into the shell logic to perform memory management when the rasterizer data structure is processed.

23. The data structure of claim 22 wherein the data structure is dynamically generated on the machine after the graphical state is detected.

24. The data structure of claim 22 wherein the data structure is prefabricated and made accessible on the machine based on the graphical state and the machine.

25. The data structure of claim 22 wherein a reference to the data structure is indexed based on the graphical state.